Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Cancelled)

2. (Currently Amended) A logic development system using an external microcomputer which replaces a built-in microcomputer incorporated in an existing electronic control unit, comprising:

a mother board including an application block and a first communication block;

a core board including one or more devices which simulate, by software, peripheral devices of the built-in microcomputer so as to execute an input or output process, the core board further including a computing block and a second communication block;

a peripheral component interconnect (PCI) bus coupling said mother board and said core board;

an interface board including a port assignment conversion board, a plurality of standard circuits, and a plurality of facility boards which are associated with hardware of said electronic control unit, said standard circuits and facility boards being selectable by said port assignment conversion board, and said port assignment conversion board being coupled to said core board via a harness; and

a bus controller in said computing block interposed between said first communication block in said mother board and each of said one or more devices, [[wherein,]]

<u>an internal memory in said computing block coupled to said bus controller over a first internal bus, wherein,</u>

said first communication block included in said mother board and said bus controller are coupled to each other over said PCI bus, and said bus controller and each of said one or more devices are coupled to each other over [[an]] <u>a second</u> internal bus, [[and]]

said first communication block and each of said one or more devices transfer data to or from each other according to two transfer techniques,

wherein a first one of the two transfer techniques transfers a first portion of the data by way of said PCI bus, bus controller, and the second internal bus[[.]] without using said internal memory, the first one of the two transfer techniques being invoked for acquiring input information on the mother board during a first action of an application, and

a second one of the two transfer techniques transfers a second portion of the data via said internal memory, wherein the second portion of the data from one of said one or more devices is stored in said internal memory via said first internal bus and said bus controller, and the second portion of the data stored in said internal memory is transmitted to said first communication block via said bus controller concurrently, wherein the second transfer technique is carried out before a second action of the application responsive to an interrupt request from the core board to the mother board.

3. (Currently Amended) A microcomputer logic development system according to claim [[1]] 2, wherein: a virtual memory device, which is provided in a memory of said core board, is interposed between said first communication block included in said mother board and said PCI bus; and when transfer data, which is transmitted over said PCI bus, is temporarily recorded in said virtual memory device in response to receiving or transmitting data, said virtual memory device behaves like a memory device included in the built-in microcomputer.

4. (Currently Amended) A microcomputer logic development system according to claim [[1]] 2, wherein: an object on which said application block acts is a vehicle; said logic development system includes an ignition switch; and when said logic development system is interlocked with an on or off state of said ignition switch, control software for said vehicle is initiated or terminated in the same manner as control software residing in said electronic control unit which is also initiated or terminated in accordance with an on or off state of said ignition switch.

5. (Previously Presented) A microcomputer logic development system according to claim 4, wherein said plurality of facility boards includes at least one facility circuit in which a microcomputer is incorporated; and said facility circuit is not actuated with the on state of said ignition switch but is actuated synchronously with starting up of the mother board.

6. (Currently Amended) A microcomputer logic development system according to claim 5, wherein said facility circuit includes a power circuit that is actuated with the on state of said ignition switch, and a logic circuit which actuates the microcomputer in the facility circuit when both an initial [[a]] signal sent from said power circuit and [[a]] an initial signal sent from said mother board become valid.

7. (Cancelled)

8. (Currently Amended) A microcomputer logic development system according to claim 4, wherein initial values to ports are set within an initialization routine which is executed on said mother board and then the state proceeds to wait the on state of said ignition switch, ~~when said ignition switch is turned on~~ after a power supply of said logic development system is turned on.

9. (Currently Amended) A microcomputer logic development system according to claim [[1]] 2, wherein: said PCI bus contains a one-channel interrupt signal line over which an interrupt request is issued from said core board to said mother board; when said interrupt signal line is activated by said core board, said application block included in said mother board accepts the interrupt request; and after the interrupt request is accepted, said interrupt signal line is inactivated.

10. (Previously Presented) A microcomputer logic development system according to claim 9, wherein when interrupt handling is terminated, said application block included in said mother board checks if said interrupt signal line is inactive.

11. (Previously Presented) A microcomputer logic development system according to claim 10, wherein when interrupt handling is terminated, if said interrupt signal line is active, said application block included in said mother board inactivates said interrupt signal line.

12. (Currently Amended) A microcomputer logic development system according to claim [[1]] 2, wherein: said computing block included in said core board includes a facility for temporarily fetching data; when a large amount of data is transferred between said mother board and each of said one or more devices included in said core board, the large amount of data is transferred in a burst mode between said mother board and said computing block, and transferred in a non-burst mode between said computing block and each of said one or more devices.

13. (Previously Presented) A microcomputer logic development system according to claim 9, wherein after said application block included in said mother board accepts an interrupt request, said application block acquires interrupt flags from each of said one or more devices over said PCI bus; after said application block acquires interrupt flags, said application block clears the interrupt flags present in each of said one or more devices.

14. (Previously Presented) A microcomputer logic development system according to claim 13, wherein after said application block included in said mother board acquires interrupt flags, said application block executes a process associated with each of the acquired interrupt flags.

15. (Previously Presented) A microcomputer logic development system according to claim 9, wherein: after said application block included in said mother board accepts an interrupt request, said application block acquires a plurality of interrupt flags from each of said one or more devices over said PCI bus; said application block selects one interrupt flag assigned a high priority, and executes a process associated with the interrupt flag; and after the process is completed, said application block clears a process completion interrupt flag present in each of said one or more devices.

16. (Previously Presented) A microcomputer logic development system according to claim 15, wherein after said application block selects one interrupt flag assigned a high priority, and executes a process associated with the interrupt flag, said application block re-acquires a plurality of interrupt flags from each of said one or more devices over said PCI bus.

17. (Currently Amended) A microcomputer logic development system according to claim 13, wherein said interrupt flags are concurrently stored ~~at successive addresses~~ in one of one or more registers included in each of said one or more devices.

18. (Previously Presented) A microcomputer logic development system according to claim 17, wherein: a plurality of core boards are included; interrupt flags representing interrupts caused by each of a plurality of resources that are included in each of said core boards are stored in a register included in each of said core boards; the interrupt flags representing interrupts caused by each of the resources included in a first one of the plurality of core boards are stored in the register included in the first one of the plurality of core boards; and an extension interrupt flag indicating whether interrupt flags, representing interrupts caused by each of the resources included in each remaining ones of the plurality of core boards, are present is stored in association with each core board.

19. (Previously Presented) A microcomputer logic development system according to claim 18, wherein if said extension interrupt flag demonstrates that interrupt flags are stored in the register included in any of the remaining core boards, said application block acquires the interrupt flags from the register in the remaining core board.

20. (Currently Amended) A microcomputer logic development system according to claim [[1]] <u>2</u>, wherein: a plurality of core boards are included; the first one of the plurality of core boards alone includes a free-run timer; said first one of the plurality of core boards includes at least resources that act synchronously with a timer value of said free-run timer; and remaining ones of the plurality of core boards include resources independent of said free-run timer.

21. (Original) A microcomputer logic development system according to claim 20, wherein: the resources that act synchronously with the timer value of said free-run timer include a comparator and a capture unit; and the resources independent of said free-run timer include a pulse-width modulator (PWM), a communication unit, an A/D converter, and ports.

22. (Cancelled)

23. (Currently Amended) A logic development method for a microcomputer including a mother board having an application block and a first communication block, a core board having one or more devices which simulate, by software, peripheral devices of the built-in microcomputer so as to execute an input or output process, the core board further including a computing block and a second communication block, an interface board having a port assignment conversion board, a plurality of standard circuits, and a plurality of facility boards which are associated with hardware of an electronic control unit, said standard circuits and facility boards being selectable by said port assignment conversion board, and said port assignment conversion board being coupled to said core board via a harness, [[and]] a peripheral component interconnect (PCI) bus over which said mother board and said core board are coupled to each other, a bus controller in said computing block interposed between said first communication block in said mother board and each of said peripheral devices, an internal memory in said computing block coupled to said bus controller over a first internal bus, said microcomputer logic development method comprising:

issuing an interrupt request from said core board to said mother board over a one-channel interrupt signal line contained in said PCI bus;

accepting the interrupt request when said interrupt signal line is activated via said core board; [[and]]

inactivating said interrupt signal line after the interrupt request is accepted;

transferring data between said first communication block and each of the peripheral devices according to two transfer techniques,

wherein, a first one of the two transfer techniques transfers a first portion of the data by way of said PCI bus, bus controller, and the second internal bus without using said internal memory, the first one of the two transfer techniques being invoked for acquiring input information on the mother board during a first action of an application, and

a second one of the two transfer techniques transfers a second portion of the data via said internal memory, wherein the second portion of the data from one of the peripheral devices is stored in said internal memory via said first internal bus and said bus controller, and the second portion of the data stored in said internal memory is transmitted to said first communication block via said bus controller concurrently, wherein the second transfer technique is carried out before a second action of the application responsive to the interrupt request from the core board to the mother board.

24. (Previously Presented) A microcomputer logic development method according to claim 23, further comprising the steps of:

after an interrupt request is accepted, acquiring interrupt flags from each of said one or more devices over said bus; and

after the interrupt flags are acquired, clearing the interrupt flags from each of said one or more devices.

25-29. Cancelled.

30. (New) The system of claim 2, wherein the one or more devices are field-programmable gate arrays, and under the first transfer technique, the first portion of the data is transferred directly between the mother board and the one or more field-programmable gate arrays via the PCI bus, bus controller, and internal bus without using said internal memory, and under the second transfer technique, the second portion of the data from the one or more field-programmable gate arrays is stored in the internal memory first before transfer of the second portion of the data to the mother board.

31. (New) The system of claim 2, wherein the application is an automotive engine control software application.